SOA Final Report

蔡诗怀
2021080068
计17

姚子其
2020080109
计07

李文赢
2020080108
计05

1 Project Overview

WEGO (Worldwide Exploration Guide Online) is an AI-powered vacation itinerary planner designed to assist users in creating high-quality travel plans effortlessly. The project aims to help individuals who struggle with planning their vacations due to a lack of knowledge about travel destinations or indecisiveness. WEGO not only generates a detailed travel plan for users, but also combines various information, such as weather forecasts and flight details, to help users make informed decisions. Furthermore, WEGO also includes an LLM-powered customer support, allowing the user to inquire about the trip and to make revisions to the plan simply using natural language.

2 Features

Our project contains the following features:

- A website for users to interact with
- Forms for users to be able to enter relevant travel details, such as location, date, interests, activity level
- Generation of personalized travel plan for the user based on the user's interests and requests
- Detailed description of each tourist attraction in the plan
- Images of the destination city and each attraction
- List of flights from departure city to destination city
- Weather forecast for each day of the trip
- Map containing the location pins of the attractions
- LLM-powered customer support, that helps the user with any inquiries, and can also adjust the itinerary if the user wishes to change some things

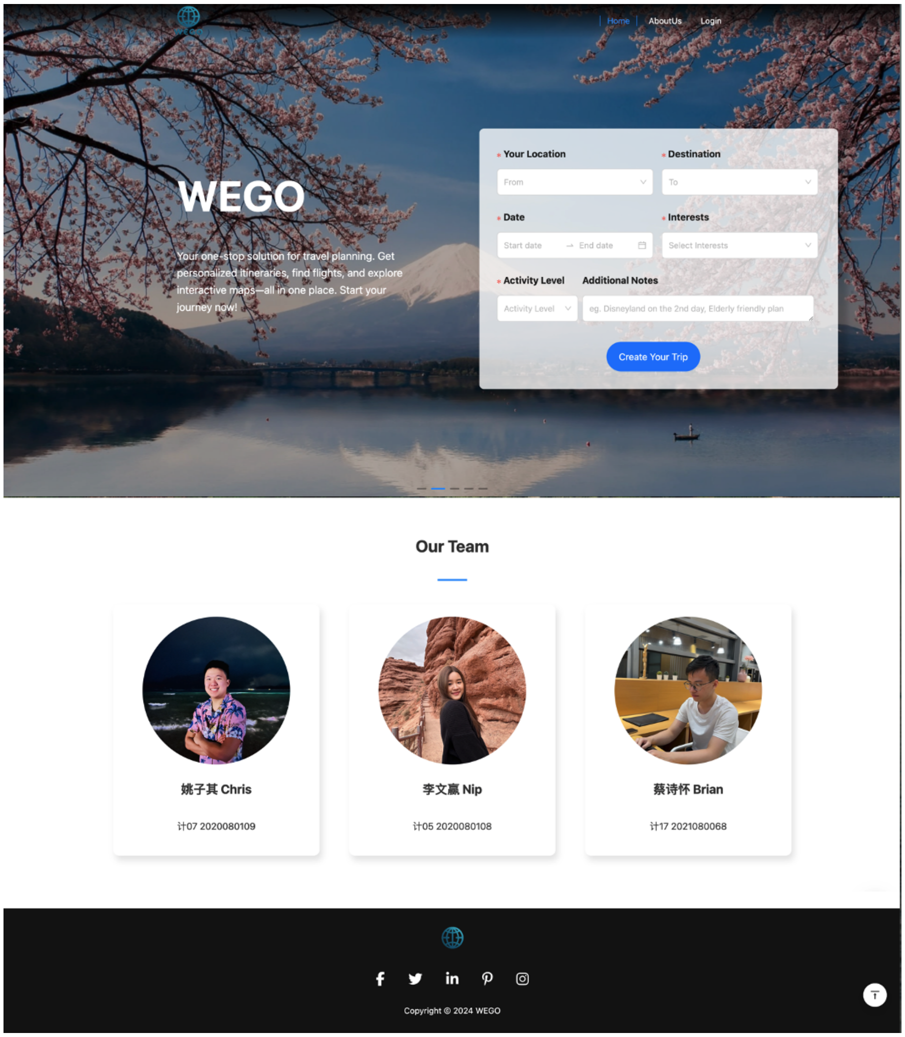
3 Pipeline

The process begins with the user entering their vacation requirements, such as location, date, and other relevant information, into the website fronted. All these data are forwarded to the website backend, which abstracts the calling and processing of various APIs. Then, the requests are sent to flight and weather APIs. These APIs return the necessary data, which will be used to generate the plan and provide suggestions for the user. The LLM is prompted based on user requests and weather data to generate a customized and detailed itinerary options. The backend parses the response and returns it to the frontend which compiles all this information to generate a comprehensive and personalized itinerary for the user. For the customer support feature, the LLM processes these prompts and provides relevant information to refine the travel plan.

4 Frontend

The frontend is developed using React with Typescript and the Ant Design framework, the website consists of two pages:

1. **Homepage**
   The first page shows introduction of WEGO, our teams members, and user input form which user can input required informations needed for generate the travel plan.



WEGO

Your one-stop solution for travel planning. Get personalized itineraries, find flights, and explore interactive maps—all in one place. Start your journey now!

* Your Location: From
* Destination: To
* Date: Start date → End date
* Interests: Select Interests
* Activity Level: Activity Level
Additional Notes: eg. Disneyland on the 2nd day, Elderly friendly plan

Our Team

姚子其 Chris
计07 2020080109

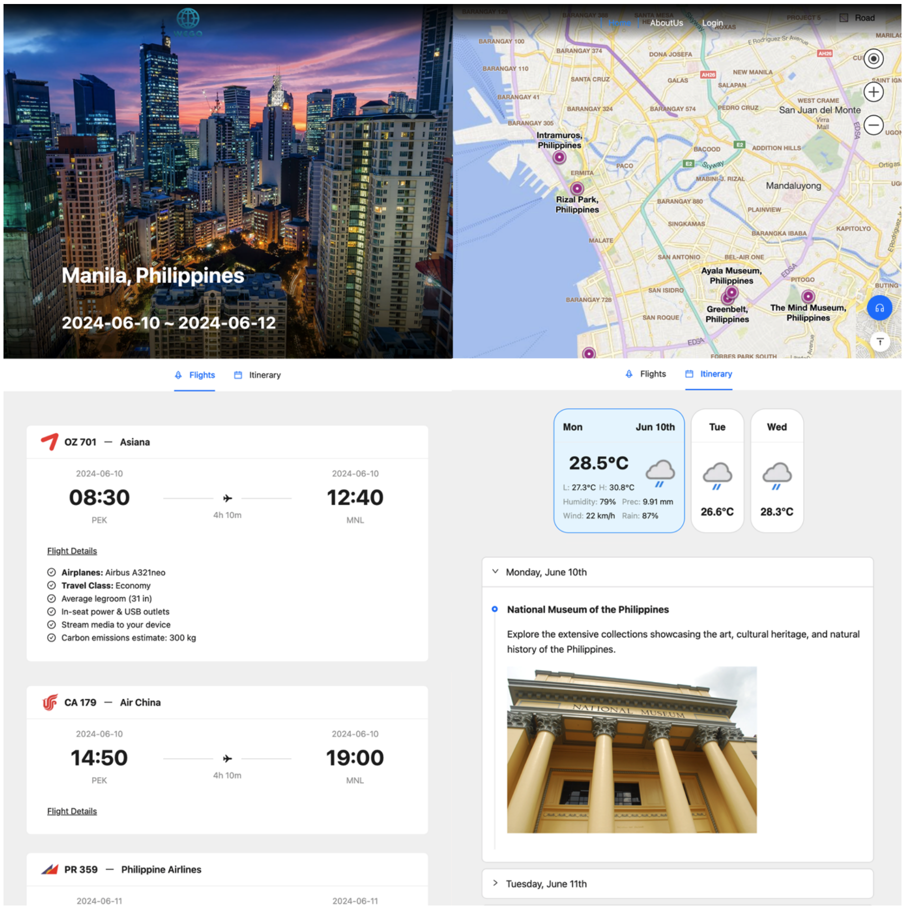
李文赢 Nip
计05 2020080108

蔡诗怀 Brian
计17 2021080068

Copyright © 2024 WEGO

Figure 1: Homepage

2. **Itinerary page**
   The result page shows detailed plan for the desired destination, including:
   - Flight details
   - Weather details
   - Itinerary details showing each day's activities with images and descriptions of attractions
   - Interactive map displaying a map with location pins for each attractions (Implemented with Bing Maps API)
   - Customer support chatbot allowing users to make inquiries and adjust their itineraries



Manila, Philippines

2024-06-10 ~ 2024-06-12

OZ 701 — Asiana

2024-06-10 08:30 PEK — 4h 10m — 2024-06-10 12:40 MNL

Flight Details

- Airplanes: Airbus A321neo
- Travel Class: Economy
- Average legroom (31 in)
- In-seat power & USB outlets
- Stream media to your device
- Carbon emissions estimate: 300 kg

CA 179 — Air China

2024-06-10 14:50 PEK — 4h 10m — 2024-06-10 19:00 MNL

Flight Details

PR 359 — Philippine Airlines

2024-06-11 — 2024-06-11

Mon Jun 10th
28.5°C
L: 27.3°C H: 30.8°C
Humidity: 79% Prec: 9.91 mm
Wind: 22 km/h Rain: 87%

Tue 26.6°C

Wed 28.3°C

Monday, June 10th

National Museum of the Philippines

Explore the extensive collections showcasing the art, cultural heritage, and natural history of the Philippines.

Tuesday, June 11th

Figure 2: Itinerary Page

5 Backend

The backend of the server is developed using Django. It has the following key components:

5.1 LLM

The backend is responsible for embedding the user requests into a prompt and forwarding the request to the LLM API. The LLM that we used for this project is GPT-4o. We manually try different prompt for the travel plan generation, and ultimately settled for a prompt template shown below. We also added a system prompt of "You are an expert travel planner.". By filling the necessary information into the prompt template, a response in JSON format can be obtained. This is parsed and returned to the frontend to be displayed. For the customer service feature, the first conversation turn (i.e., the filled prompt template and

```
Generate a {day} day travel plan {intensity} for a person traveling to {destination}. The
person is interested in {interests}.{custom_request}{weather} You are to give a travel
plan in JSON format like the following:
{{ "Day 1" : [ {{ "Location" : "Forbidden City", "Activity" : "Visit the historical
palaces, halls, and courtyards and learn about rich history of China's imperial past" }},
{{ "Location" : "<location>", "Activity" : "<activity>" }} ], "Day 2" : [ {{ "Location":
"<location>", "Activity": "<activity>", ...] }}, ...}}
```

Figure 3: Prompt Template for Travel Plan Generation

LLM's response) is added into the messages parameter of the chat.completions api of OpenAI API together with the user query. This provides the model with the sufficient context to generate helpful responses and to revise the plan if necessary. The generated response is parsed and returned to the frontend to update the displayed content.

5.2 Weather, Flights, Image APIs

For the weather forecast, we used weatherapi.com to obtain weather data on the days covered in the duration of the trip.

For the flight details, we used the google flights api provided by serpapi.com to obtain the best flights for a given user query.

For the images, we wrote a python script using the duckduckgo library in order to search for, and filter through images for the destination location, and local attractions.

6 Work Distribution

蔡诗怀:

- All LLM-related features in the backend, including prompt engineering, pre-processing and post-processing
- Weather feature
- Frontend deployment

姚子其:

- Miscellaneous Backend APIs such as images and flights
- Backend deployment
- Frontend Map feature

李文赢:

- Frontend Design